UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/733,037	12/11/2003	Keith J. Purcell	RSW920030159US1	8444

46320 7590 09/14/2010
CAREY, RODRIGUEZ, GREENBERG & PAUL, LLP
STEVEN M. GREENBERG
950 PENINSULA CORPORATE CIRCLE
SUITE 2022
BOCA RATON, FL 33487

EXAMINER	
CHEN, QING	

ART UNIT	PAPER NUMBER
2191	

MAIL DATE	DELIVERY MODE
09/14/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/733,037
Filing Date: December 11, 2003
Appellant(s): PURCELL, KEITH J.

Steven M. Greenberg (Reg. No. 44,725)
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed on July 19, 2010 appealing from the Office action mailed on January 19, 2010.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The Examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The Examiner agrees with the statement of the status of claims contained in the brief.

(4) Status of Amendments After Final

The Examiner agrees with the statement of the status of amendments contained in the brief.

(5) Summary of Claimed Subject Matter

The Examiner agrees with the summary of claimed subject matter contained in the brief.

(6) Grounds of Rejection to be Reviewed on Appeal

The Examiner agrees with the statement of the grounds of rejection to be reviewed set forth in the brief.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

2004/0088688	HEJLSBERG	5-2004
7,185,046	FERSTL	2-2007

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

1. **Claims 1, 4-9, 12-19, and 21-26** are rejected under 35 U.S.C. 103(a) as being unpatentable over **US 2004/0088688 (hereinafter "Hejlsberg")** in view of **US 7,185,046 (hereinafter "Ferstl")**.

As per **Claim 1**, Hejlsberg discloses:

- receiving from an author over a computer communications network a description of a computing application in a web service executing in memory by a processor in a computer *(see Figures 1 and 2; Paragraph [0006], "... a file, such as a database mapping description or declaration, is authored by a user or a design tool in a particular data language in which a format can be defined, such as XML. Such an exemplary file is referred to as a blueprint ..."; Paragraph [0017], "Blueprints allow the ASP.NET markup-and-code paradigm to be extended to other domains such as user interfaces, database mapping, web services, and compiled*

extensible stylesheet language (XSL) transforms."; Paragraph [0087], "The present invention can be applied to a wide variety of technologies, such as ... web services ...");

- parsing said description in said web service to identify object parameters for said computing application *(see Paragraph [0035], "Upon receiving the blueprint 200, the blueprint translator 210 parses the blueprint (using, e.g., an XML parser) ..."; Paragraph [0047], "In addition, the framework defines a file extension, .dbml, and includes a blueprint translator that can translate .dbml files containing XML-formatted mapping descriptions into source code that targets the framework. An exemplary .dbml blueprint is generated by a user or a design tool and is shown below.")*;

- supplying said description to a node *(see Paragraph [0035], "... provides the parsed blueprint to a Document Object Model (DOM) for further processing. The output of the DOM is provided to a semantic analyzer and code generator. Source code 220 is thereby generated in accordance with predetermined schemas, patterns, and/or hierarchical rules, for example.")*;

- applying said description to a located coding module to generate at least one output object corresponding to the identified object parameters *(see Paragraph [0035], "... provides the parsed blueprint to a Document Object Model (DOM) for further processing. The output of the DOM is provided to a semantic analyzer and code generator. Source code 220 is thereby generated in accordance with predetermined schemas, patterns, and/or hierarchical rules, for example."; Paragraph [0058], "... a blueprint translator can use the CodeDOM (an object model for abstract syntax trees and code generation provided in the System.CodeDom namespace) to generate source code in a language-neutral fashion.")*; and

- returning said at least one output object to the author over the computer communications network *(see Paragraph [0035], "The source code 220 may access or point to a supporting framework or class library 230.")*.

However, Hejlsberg does not disclose:

- locating a coding module corresponding to at least one of the object parameters within a node contained within a computational grid coupled to the web service over a computer communications network, the computational grid comprising a plurality of computers sharing computational resources, said computational grid further comprising a plurality of coding modules.

Ferstl discloses:

- locating a coding module corresponding to at least one of object parameters within a node contained within a computational grid coupled to a web service over a computer communications network, the computational grid comprising a plurality of computers sharing computational resources, said computational grid further comprising a plurality of coding modules *(see Column 1: 52-59, "A computing grid is a hardware and software infrastructure serving to handle computing jobs submitted by a user. The computing grid may interconnect distributed computers, storage devices, mobile devices, instruments, sensors, data bases and/or software applications. Generally a computing grid may comprise virtually any kind of computing device and includes a grid infrastructure to handle the distribution of computing jobs." and 65-67 to Column 2: 1-8, "Upon receiving an instruction to distribute a computing job the grid infrastructure selects a suitable computing device and transfers the computing job to the selected computing device." and "Accordingly, a user or application at a client device may issue an*

instruction to execute a computing job towards the grid infrastructure which in turn selects a suitable processing element and the processing results are ultimately returned to the client."; Column 12: 25-31, "In an example, the selection section obtains the selection information and accesses data specifying corresponding features of a plurality of job handlers, for example, in a configuration file specifying features of a plurality of job handlers available. Then, the selection section may identify a suitable job handler matching the selection information in association with the job request.").

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Ferstl into the teaching of Hejlsberg to modify Hejlsberg's invention to include locating a coding module corresponding to at least one of the object parameters within a node contained within a computational grid coupled to the web service over a computer communications network, the computational grid comprising a plurality of computers sharing computational resources, said computational grid further comprising a plurality of coding modules. The modification would be obvious because one of ordinary skill in the art would be motivated to utilize a variety of computing devices and/or software applications to quickly solve a single computing task *(see Ferstl – Column 1: 46-50).*

As per **Claim 4**, the rejection of **Claim 1** is incorporated; and Hejlsberg further discloses:

- wherein said description is generated using Object Meta Language (OML) *(see Paragraph [0006], "... a file, such as a database mapping description or declaration, is authored by a user or a design tool in a particular data language in which a format can be defined, such as XML. Such an exemplary file is referred to as a blueprint ...").*

As per **Claim 5**, the rejection of **Claim 4** is incorporated; and Hejlsberg further discloses:

- wherein said OML is an eXtensible Markup Language (XML) dialect *(see Paragraph [0006], "... a file, such as a database mapping description or declaration, is authored by a user or a design tool in a particular data language in which a format can be defined, such as XML. Such an exemplary file is referred to as a blueprint ...").*

As per **Claim 6**, the rejection of **Claim 1** is incorporated; and Hejlsberg further discloses:

- wherein said located coding module is an XML template *(see Paragraph [0047], "In addition, the framework defines a file extension, .dbml, and includes a blueprint translator that can translate .dbml files containing XML-formatted mapping descriptions into source code that targets the framework.").*

As per **Claim 7**, the rejection of **Claim 1** is incorporated; and Hejlsberg further discloses:

- wherein said located coding module is an eXtensible Stylesheet Language (XSL) style sheet *(see Paragraph [0017], "Blueprints allow the ASP.NET markup-and-code paradigm to be extended to other domains such as user interfaces, database mapping, web services, and compiled extensible stylesheet language (XSL) transforms.").*

As per **Claim 8**, the rejection of **Claim 7** is incorporated; and Hejlsberg further discloses:

- parsing said description to locate at least one variable *(see Paragraph [0048], "... mapping the Customers table in the database to a Customer class in the Northwind namespace.*

Further details of the mapping include the CustomerID column that maps to an Id property, the ContactName column that maps to a Name property, etc."); and

- substituting said at least one variable with at least one replacement variable, wherein said at least one replacement variable is the result of an XML/XSL transform *(see Paragraph [0048], "... the blueprint calls for an Orders collection to be generated in the Customer class based on the relation between the Customer and Order classes described in the <relation> element."; Paragraph [0050], "A blueprint like the one set forth above would typically be generated by a database design tool, but it could also be authored manually or created by an XML transformation.").*

As per **Claim 9**, the rejection of **Claim 6** is incorporated; and Hejlsberg further discloses:

- parsing said description to locate at least one variable *(see Paragraph [0048], "... mapping the Customers table in the database to a Customer class in the Northwind namespace. Further details of the mapping include the CustomerID column that maps to an Id property, the ContactName column that maps to a Name property, etc.")*; and

- substituting said at least one variable with at least one replacement variable, wherein said at least one replacement variable is stored in said XML template *(see Paragraph [0048], "... the blueprint calls for an Orders collection to be generated in the Customer class based on the relation between the Customer and Order classes described in the <relation> element."; Paragraph [0050], "A blueprint like the one set forth above would typically be generated by a database design tool, but it could also be authored manually or created by an XML transformation.").*

Claims 12-18 are computer program product claims corresponding to the method claims above (Claims 1 and 4-9) and, therefore, are rejected for the same reasons set forth in the rejections of Claims 1 and 4-9.

As per **Claim 19**, Hejlsberg discloses:

- a web service for receiving an application description from an author from over a computer communications network, for parsing said application description to identify object parameters for a computing application, to locate a coding module corresponding to at least one of the object parameters within a node, for supplying said application description to said node in which said located coding module applies said application description to generate at least one output object corresponding to the identified object parameters in said application description, and returning said at least one output object to the author over the computer communications network *(see Figures 1 and 2; Paragraph [0006], "... a file, such as a database mapping description or declaration, is authored by a user or a design tool in a particular data language in which a format can be defined, such as XML. Such an exemplary file is referred to as a blueprint ..."; Paragraph [0017], "Blueprints allow the ASP.NET markup-and-code paradigm to be extended to other domains such as user interfaces, database mapping, web services, and compiled extensible stylesheet language (XSL) transforms."; Paragraph [0035], "Upon receiving the blueprint 200, the blueprint translator 210 parses the blueprint (using, e.g., an XML parser), and provides the parsed blueprint to a Document Object Model (DOM) for further processing. The output of the DOM is provided to a semantic analyzer and code generator.*

Source code 220 is thereby generated in accordance with predetermined schemas, patterns, and/or hierarchical rules, for example. The source code 220 may access or point to a supporting framework or class library 230."; Paragraph [0047], "In addition, the framework defines a file extension, .dbml, and includes a blueprint translator that can translate .dbml files containing XML-formatted mapping descriptions into source code that targets the framework. An exemplary .dbml blueprint is generated by a user or a design tool and is shown below."; Paragraph [0087], "The present invention can be applied to a wide variety of technologies, such as ... web services ..."; Paragraph [0058], "... a blueprint translator can use the CodeDOM (an object model for abstract syntax trees and code generation provided in the System.CodeDom namespace) to generate source code in a language-neutral fashion.").

However, Hejlsberg does not disclose:

- a computational grid, wherein said computational grid includes a plurality of computers sharing computational resources, said computational grid comprising a plurality of nodes, each node comprising at least one programming model.

Ferstl discloses:

- a computational grid, wherein said computational grid includes a plurality of computers sharing computational resources, said computational grid comprising a plurality of nodes, each node comprising at least one programming model *(see Column 1: 52-67 to Column 2: 1-8, "A computing grid is a hardware and software infrastructure serving to handle computing jobs submitted by a user. The computing grid may interconnect distributed computers, storage devices, mobile devices, instruments, sensors, data bases and/or software applications. Generally a computing grid may comprise virtually any kind of computing device and includes a*

grid infrastructure to handle the distribution of computing jobs." and "Upon receiving an instruction to distribute a computing job the grid infrastructure selects a suitable computing device and transfers the computing job to the selected computing device." and "Accordingly, a user or application at a client device may issue an instruction to execute a computing job towards the grid infrastructure which in turn selects a suitable processing element and the processing results are ultimately returned to the client.").

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Ferstl into the teaching of Hejlsberg to modify Hejlsberg's invention to include a computational grid, wherein said computational grid includes a plurality of computers sharing computational resources, said computational grid comprising a plurality of nodes, each node comprising at least one programming model. The modification would be obvious because one of ordinary skill in the art would be motivated to utilize a variety of computing devices and/or software applications to quickly solve a single computing task *(see Ferstl – Column 1: 46-50).*

As per **Claim 21**, the rejection of **Claim 19** is incorporated; and Hejlsberg further discloses:

- wherein said application description is generated using Object Meta Language (OML) *(see Paragraph [0006], "... a file, such as a database mapping description or declaration, is authored by a user or a design tool in a particular data language in which a format can be defined, such as XML. Such an exemplary file is referred to as a blueprint ...").*

As per **Claim 22**, the rejection of **Claim 21** is incorporated; and Hejlsberg further discloses:

- wherein said OML is an eXtensible Markup Language (XML) dialect *(see Paragraph [0006], "... a file, such as a database mapping description or declaration, is authored by a user or a design tool in a particular data language in which a format can be defined, such as XML. Such an exemplary file is referred to as a blueprint ...").*

As per **Claim 23**, the rejection of **Claim 19** is incorporated; and Hejlsberg further discloses:

- wherein said coding modules are XML templates *(see Paragraph [0047], "In addition, the framework defines a file extension, .dbml, and includes a blueprint translator that can translate .dbml files containing XML-formatted mapping descriptions into source code that targets the framework.").*

As per **Claim 24**, the rejection of **Claim 19** is incorporated; and Hejlsberg further discloses:

- wherein said coding modules are eXtensible Stylesheet Language (XSL) style sheets *(see Paragraph [0017], "Blueprints allow the ASP.NET markup-and-code paradigm to be extended to other domains such as user interfaces, database mapping, web services, and compiled extensible stylesheet language (XSL) transforms.").*

As per **Claim 25**, the rejection of **Claim 24** is incorporated; and Hejlsberg further discloses:

- parsing said description to locate at least one variable *(see Paragraph [0048], "... mapping the Customers table in the database to a Customer class in the Northwind namespace. Further details of the mapping include the CustomerID column that maps to an Id property, the ContactName column that maps to a Name property, etc.")*; and

- substituting said at least one variable with at least one replacement variable, wherein said at least one replacement variable is the result of an XML/XSL transform *(see Paragraph [0048], "... the blueprint calls for an Orders collection to be generated in the Customer class based on the relation between the Customer and Order classes described in the <relation> element."; Paragraph [0050], "A blueprint like the one set forth above would typically be generated by a database design tool, but it could also be authored manually or created by an XML transformation.").*

As per **Claim 26**, the rejection of **Claim 23** is incorporated; and Hejlsberg further discloses:

- parsing said description to locate at least one variable *(see Paragraph [0048], "... mapping the Customers table in the database to a Customer class in the Northwind namespace. Further details of the mapping include the CustomerID column that maps to an Id property, the ContactName column that maps to a Name property, etc.")*; and

- substituting said at least one variable with at least one replacement variable, wherein said at least one replacement variable is stored in said XML template *(see Paragraph [0048],*

"... the blueprint calls for an Orders collection to be generated in the Customer class based on the relation between the Customer and Order classes described in the <relation> element."; Paragraph [0050], "A blueprint like the one set forth above would typically be generated by a database design tool, but it could also be authored manually or created by an XML transformation.").

(10) Response to Argument

THE REJECTION OF CLAIMS 1, 4-9, 12-19 AND 21-26 UNDER 35 U.S.C. § 103

In the Appeal Brief, Appellant argues:

a) In particular, Examiner argues at page 7 of the Last Non-Final Office Action that column 1, line 52-59 and 65-67, column 2, lines 1-8 and column 12, lines 25- 31 provide the relevant teaching. Examiner specifically states,

> However, Hejlsberg does not disclose:
>
> locating a coding module corresponding to at least one of the object parameters within a node contained within a computational grid coupled to the web service over a computer communications network, the computational grid comprising a plurality of computers sharing computational resources, said computational grid further comprising a plurality of coding modules.
>
> Ferstl discloses:
>
> locating a coding module corresponding to at least one of object parameters within a node contained within a computational grid coupled to a web service over a computer communications network, the computational grid comprising a plurality of computers sharing computational resources, said computational grid further comprising a

> plurality of coding modules *(see Column 1: 52-59, "A computing grid is" a hardware and software infrastructure serving to handle computing jobs submitted by a user. The computing grid may interconnect distributed computers, storage devices, mobile devices, instruments, sensors, data bases and/or software applications. Generally a computing grid may comprise virtually any kind of computing device and includes a grid infrastructure to handle the distribution of computing jobs." and 65-67 to Column 2: 1-8, "Upon receiving an instruction to distribute a computing job the grid infrastructure selects a suitable computing device and transfers the computing job to the selected computing device." and "Accordingly, a user or application at a client device may issue an instruction to execute a computing job towards the grid infrastructure which in turn selects a suitable processing element and the processing results are ultimately returned to the client. "; Column 12: 25-31, "In an example, the selection section obtains the selection information and accesses data specifying corresponding features of a plurality of job handlers, for example, in a configuration file specifying features of a plurality of job handlers available. Then, the selection section may identify a suitable job handler matching the selection information in association with the job request. ").*

Thus, Examiner's analysis appears to be an aggregation of the verbatim claim language of Appellants' claim 1 and the verbatim text of the cited portions of Ferstl without any significant analysis of how Examiner has construed critical claim limitations such as "coding module" or "locate" or "object parameters", and how Examiner has applied the teachings of Ferstl to the construed claim language.

To Examiner's credit, however, on page 17 of the Last Non-Final Office Action, after repeating again verbatim Appellants' claim language of the claim limitation in dispute and the text of Ferstl. In this regard, Examiner states at page 17 that a computing grid is both hardware and software interconnecting, among other things, software applications and that "the computing grid selects a suitable processing element to process a computing job". Regrettably, as for the ONLY substantive analysis provided by Examiner, Examiner provided NO discussion of a teaching in Ferstl that directly maps to

the location of a coding module corresponding to one or more object parameters within a node contained within a computational grid coupled to a web service as claimed by Appellants in each of claims 1, 12 and 19.

(See Appeal Brief – page 8 to page 10.)

Examiner's response:

a) Examiner disagrees. Appellant's arguments are not persuasive for at least the following reasons:

First, without acquiescing to the Appellant's assertion that the Examiner provided no discussion of a teaching in Ferstl that directly maps to the location of a coding module corresponding to one or more object parameters within a node contained within a computational grid coupled to a web service, the Examiner first submits that the combination of Hejlsberg and Ferstl teaches a coding module within a node contained within a computational grid. Hejlsberg discloses a coding module *(see Figure 2: 210; Paragraph [0035], "... a semantic analyzer and code generator.")*. However, Hejlsberg does not disclose that the coding module is contained within a computational grid. Ferstl discloses coding modules contained within a computational grid *(see Column 1: 52-59, "A computing grid is a hardware and software infrastructure serving to handle computing jobs submitted by a user (emphasis added). The computing grid may interconnect distributed computers, storage devices, mobile devices, instruments, sensors, data bases and/or software applications [coding modules].")*. Thus, in view of the teaching of Ferstl, one of ordinary skill in the art would be motivated to implement Hejlsberg's semantic analyzer and code generator of the blueprint translator as a node of a hardware/software infrastructure of a

computational grid. Such modification would provide the benefit of utilizing a variety of computing devices and/or software applications to quickly solve a single computing task *(see Ferstl – Column 1: 46-50)*.

Second, with respect to the Appellant's assertion that the Examiner provided no discussion of a teaching in Ferstl that directly maps to the location of a coding module corresponding to one or more object parameters within a node contained within a computational grid coupled to a web service, as previously pointed out in the Final Rejection (mailed on 01/19/2010) and further clarified hereinafter, the Examiner respectfully submits that Ferstl clearly discloses "locating a coding module corresponding to at least one of object parameters within a node contained within a computational grid coupled to a web service over a computer communications network" *(see Column 1: 52-59, "A computing grid is a hardware and software infrastructure serving to handle computing jobs submitted by a user (emphasis added). The computing grid may interconnect distributed computers, storage devices, mobile devices, instruments, sensors, data bases and/or software applications (emphasis added)."; Column 8: 56-67 to Column 9: 1-10, "The selection section may then select [locate] one of the job handler candidates [coding module ... within a node] for executing the computing job ... In an example, this further information includes obtaining offers from each of the candidate job handlers indicating system characteristics of the job handlers [coding module corresponding to at least one of object parameters]. The characteristics may include information on a memory capacity available; a number of processing units available; available software licenses; special resources such as graphic rendering resources, a type of operations available at the job handler, a security level available, and similar [at least one of object parameters]."; Column 10: 61-64, "Further,*

according to an example the computing grid also includes a plurality of web servers [web services] ... for enabling handling of computing jobs."). Note that Ferstl discloses that a computing grid is a hardware and software infrastructure interconnecting distributed computers, storage devices, mobile devices, instruments, sensors, data bases and/or software applications. Ferstl also discloses that the computing grid selects a job handler candidate to execute a computing job. Thus, one of ordinary skill in the art would readily comprehend that the selected job handler candidate may be a software application (*i.e.*, coding module) contained within a computing node that can satisfy an object parameter of the computing job.

Therefore, for at least the reasons set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 1, 12, and 19 are proper and therefore, maintained.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the Examiner in the Related Appeals and Interferences section of this Examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Qing Chen

/Q. C./

Examiner, Art Unit 2191

Conferees:

/Wei Y Zhen/

Supervisory Patent Examiner, Art Unit 2191

/Lewis A. Bullock, Jr./
Supervisory Patent Examiner, Art Unit 2193